SELinux

Security Enhenced Linux

进程以某个用户的身份来执行

例如对于互联网提供服务的httpd进程被控制将会。。。

rm /etc 并不需要有/etc目录的执行权限

为了控制这方面的权限与进程的问题，所以NSA着手对这方面的控管。其实SELinux就是在进行程序、文件等权限设置依据的一个内核模块。控制网络服务能否访问系统资源的一道关卡。

DAC 自主访问控制

完全基于文件的权限。当用户取得权限之后他可以通过这个进程和自己的默认权限来处理自己的文件资源。这很可能会有资源误用的问题

**MAC 强制访问控制** 最小权限集合的理念

他可以针对特定的**进程**与特定的**文件资源**来进行权限的控制，也就是说你是root但是获得的权限不一定是root，而要看该进程的设置而定。这样我们**控制的主体就是进程而不是用户。**此外，这个主体进程也不能任意使用系统文件资源，因为每一个文件系统资源也有针对该主体进程设置可取用的权限！这样控制的项目就细多了，但是系统进程和文件那么多，SELinux也有默认的一些策略，并在该策略内提供多个规则，让你可以选择是否启用该控制规则。

这样进程能够活动的空间就小了，例如默认httpd就只能/var/www下访问文件了，去其他目录访问数据时除了**规则设置**开放外，**目标目录**也要设置成httpd**可读取的模式**才行，限制非常多，所以httpd被控制了他也无权浏览shadow文件。

ls -Z 查看文件的selinux属性标签

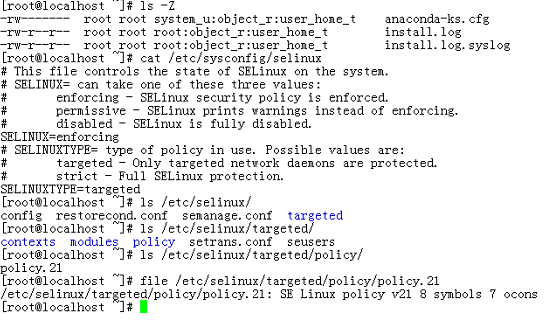
ps auxZ

selinux主配置文件：(这个和让彻底关闭时候的那个配置文件有什么联系？)

/etc/sysconfig/selinux

selinux 策略规则库，编译为二进格式. 在：

/etc/selinux/targeted/policy/ 下有一个名叫policy.21的文件



#getenforce 查看selinux状态：

permissive 仅记录，不采取行动

enforcing

日志记录在/var/log/audit下，任何违反的访问都记录在这里。

**彻底关闭**selinux /etc/selinux/sonfig中：

SELINUX=disabled 重启生效

**暂时关闭selinux**

setenforce 0

根目录下也有一个seliunx文件，这是一个伪文件，也是内核接口。

ls /selinux/

echo “1” > /selinux/enforce 其实就是#setenforce

改变文件标签：

chcon ：change file selinux security context

-t 改变标签类型

-u selinux的用户

-R 递归，正常修改不改文件夹内的

--reference=RFILE

use RFILE’s context instead of using a CONTEXT value

例如：

touch /tmp/a.txt

ls -Z /tmp

ls -Z /var/log

chron -t var\_log\_t /tmp/a.txt

ls -Z/tmp

不同目录的创建标签不同

例如httpd安装后创建的标签为xxx运行时就只允许访问标签为此的文件，在其他建立的标签不同的不允许





即可访问bbs目录 改变标签开放服务

默认samba在selinux开启状态下无法写入，如果将标签改为samba\_share\_t即可上传， 

但家目录如果也想共享不能这样做，别人也就能访问了.此时需要运行时配置：

**booleans布尔值**：不同服务的不同开关有不同的布尔值。

运行时配置：

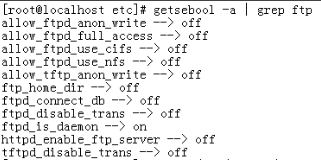
**getsebool**  查看布尔值

-a 所有

例：**setseboo**l samba\_enable\_home\_dirs=0(或者on)

修改布尔值打开或者关闭某项

同样vsftps也接受



如果想要实现selinux开启匿名用户目录上传就需要把第一个改为on

恢复为默认类型：

#restorecon 常用-R

恢复默认的上下文

#semanage

se策略管理工具 不重新编译就重新改变策略

man一下,例如：

semanage fcontext -l系统上当前默认的文件的默认标签

semanage fcontext -l | grep /var/www

semanage port -l | grep80

semanage user -l

-a等 man一下自己看 有例子

一、selinux简介

SELinux is a software product that includes several mechanisms that protect against attacks exploiting software vulnerabilities.

SELinux implements role-based access control and sandboxing.

SELinux also provides a logging and audit facility that records attempts to exceed specified permissions.

SELinux is designed to protect against misuse and unauthorized use such as:

Unauthorized reading of data and programs

Unauthorized modification of data and programs

Bypassing application security mechanisms

Interfering with other processes

Privilege escalation

Information security breaches

二、How SELinux Works

SELinux works by associating each program or process with a sandbox known as a domain. Each domain is assigned a set of permissions sufficient to enable it to function properly but do nothing else. To enable specification of such permissions, each file is labeled with information called a security context. The definition of a domain spells out what operations it can perform on files having specific security contexts. A domain cannot access files having security contexts other than those for which it is explicitly granted access.

Under specified conditions, a process that executes a program leaves its current domain and transitions to a new domain. Typically, transitions occur upon executing a program designated as an entry point to the new domain. The new domain may have more or fewer privileges than the original domain. Thus, programs can initiate other programs having more or fewer privileges than themselves.

An SELinux facility known as type enforcement (TE) ensures that the rules governing domains are always observed. SELinux also has a secondary facility known as role-based access control (RBAC). RBAC limits user access to domains. For instance, some domains are defined to be accessible only to the system administrator, whereas other domains are defined to be publicly available to any user.

An alternative product providing functions generally similar to those of SELinux is GRSecurity

三、selinux history

The TCSEC(Trusted Computer System Evaluation Criteria) defined six evaluation classes with progressively more stringent security requirements: C1, C2, B1, B2, B3, and A1. Class C1 and C2 systems.

During the 1990s, researchers at the U.S. National Security Agency (NSA) worked with Secure Computing Corporation (SCC) to develop a strong and flexible mandatory access control architecture.

四、 Subject and Object

At its root, the SELinux security model encompasses three elements:

**Subjects**

**Objects**

**Actions**

**Objects:**

**Directories**

**File descriptors**

**Files**

**Filesystems**

**Links**

**Processes**

**Special files of various types** (block device, character device, socket, FIFO, and so on)

**Actions:**

Append

Create

Execute

Get attribute

I/O control

Link

Lock

Read

Rename

Unlink

Write

五、Security Contexts 安全上下文字段：

User **identity 身份标识**

The user identity indicates the SELinux user account associated with a subject or object. In the case of a subject, the user identity gives the SELinux user account under which the process is running. In the case of an object, the user identity gives the user account that owns the object.

**Role 角色**

Under SELinux, users are authorized to enter one or more roles, each of which defines a set of permissions a user can be granted. At a given time, a user can reside in only a single role. A user can transition from one authorized role to another by using the special command newrole. This command changes the user's SELinux role similar to the way the Linux su command changes a user's Linux identity. SELinux establishes a special role, sysadm\_r, used for administering SELinux facilities.

**Type 类型**

Types, which are also known as domains, divide subjects and objects into related groups. Types are the primary security attribute SELinux uses in making authorization decisions. They establish the sandboxes that constrain processes and prevent privilege escalation. Therefore, you can think of a type as naming a related sandbox.

六、SELinux Architecture

Kernel-level code

monitors system activity and ensures that requested operations are authorized under the currently configured SELinux policy, disallowing any operations not expressly authorized

The SELinux shared library

Most non-kernel SELinux components are linked against an SELinux shared library, currently named libselinux1.so.

A security policy

When an SELinux system starts up, it loads the local security policy from a binary policy file

Tools

SELinux commands

chcon

getenforce

setenforce

Labeled SELinux filesystems (optional)

七、Using and Administering SELinux

Permissive mode

Rebuilding policies

Labeling files

Routine system administration (changing roles, adding users, and checking file contexts)

Monitoring SELinux through log files

Miscellaneous troubleshooting

1、

echo "1" > /selinux/enforce

echo "0" > /selinux/enforce

The SELinux policy can include conditional rules that are enabled or disabled based on the current values of a set of policy booleans. These policy booleans allow runtime modification of the security policy without having to load a new policy.

The policy defines a default value for each boolean, typically false. These default values can be overridden via local settings created via the setsebool(8) utility, using -P to make the setting persistent across reboots.

semanage is used to configure certain elements of SELinux policy without requiring modification to or recompilation from policy sources.